# A Convolutional Neural Network for the Detection of Asynchronous Steady State Motion Visual Evoked Potential

Xin Zhang, Guanghua Xu, *Member, IEEE*, Xiang Mou, Aravind Ravi, Min Li, *Member, IEEE*, Yiwen Wang, *Senior Member, IEEE*, and Ning Jiang, *Senior Member, IEEE*

***Abstract*— A key issue in brain-computer interface (BCI) is the detection of intentional control (IC) states and non-intentional control (NC) states in an asynchronous manner. Furthermore, for steady-state visual evoked potential (SSVEP) BCI systems, multiple states (sub-states) exist within the IC state. Existing recognition methods rely on a threshold technique, which is difficult to realize high accuracy, i.e., simultaneously high true positive rate and low false positive rate. To address this issue, we proposed a novel convolutional neural network (CNN) to detect IC and NC states in a SSVEP-BCI system for the first time. Specifically, the steady-state motion visual evoked potentials (SSMVEP) paradigm, which has been shown to induce less visual discomfort, was chosen as the experimental paradigm. Two processing pipelines were proposed for the detection of IC and NC states. The first one was using CNN as a multi-class classifier to discriminate between all the states in IC and NC state (FFT-CNN). The second one was using CNN to discriminate between IC and NC states, and using canonical correlation analysis (CCA) to perform classification tasks within the IC (FFT-CNN-CCA). We demonstrated that both pipelines achieved a significant increase in accuracy for low-performance healthy participants when traditional algorithms such as CCA threshold were used. Furthermore, the FFT-CNN-CCA pipeline achieved better performance than the FFT-CNN pipeline based on the stroke patients' data. In summary, we showed that CNN can be used for robust detection in an asynchronous SSMVEP-BCI with great potential for out-of-lab BCI applications.**

***Index Terms*— Asynchronous brain-computer interface, convolutional neural network, steady-state motion visual evoked potentials, brain-computer interface.**

Manuscript received November 1, 2018; revised January 29, 2019 and March 27, 2019; accepted May 1, 2019. Date of publication May 7, 2019; date of current version June 6, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 51775415 and Grant 51505363, in part by the China Scholarship Council, and in part by the Early Researcher Award from the Ministry of Research, Innovation and Science of Ontario under Grant ER17-13-183. *(Corresponding authors: Guanghua Xu; Ning Jiang.)*

X. Zhang and M. Li are with the School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: zx2929108zx@stu.xjtu.edu.cn; min.li@mail.xjtu.edu.cn).

G. Xu is with the School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an 710049, China, and also with the State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: ghxu@mail.xjtu.edu.cn).

X. Mou is with the Department of Rehabilitation, Xijing Hospital, The Fourth Military Medical University, Xi'an 710032, China (e-mail: pro.mu@fmmu.edu.cn).

A. Ravi and N. Jiang are with the Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L3G1, Canada (e-mail: aravind.ravi@uwaterloo.ca; ning.jiang@uwaterloo.ca).

Y. Wang is with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong, and also with the Department of Chemical and Biological Engineering, Hong Kong University of Science and Technology, Hong Kong (e-mail: eewangyw@ust.hk).

Digital Object Identifier 10.1109/TNSRE.2019.2914904

## I. Introduction

BRAIN computer interfaces (BCI) can enable people to use a computer or other equipment without using regular output pathways of peripheral nerves and muscles [1]. Several electroencephalographic (EEG) signals are mainly used in BCI such as motor imagery, movement-related cortical potential, P-300, steady-state visual evoked potential (SSVEP) and so on. Particularly, SSVEP is evoked by periodic visual stimulation with a stationary distinct spectrum with characteristic SSVEP peaks in EEG recording over visual cortex [2]. And SSVEP-based BCIs have advantages of high information transfer rate (ITR) and no need for subject training [3]. Recently, Yan *et al.* [4] designed four novel stimulus paradigms based on basic motion modes and demonstrated that any stimulus paradigms with periodic motion can induce steady-state motion visual evoked potential (SSMVEP). It was shown that SSMVEP paradigms had low-adaptation characteristic and less visual discomfort for BCI applications.

One of the key features for a BCI to be practical applications is the ability to operate in an asynchronous mode, in which the BCI can detect whether or not the user intends to send a control command, *i.e.* a brain switch [5], [6]. To this end, the ability of any BCIs to discriminate intentional control (IC) and non-intentional control (NC) states accurately based on ongoing EEG data is highly important [7]. In the literature, true positive rate (TPR), false positive rate (FPR) and accuracy had been used to quantify the performance of an asynchronous BCI. As in any detection problem, one would like to obtain a good balance between TPR, and FPR [8].

For asynchronous SSVEP-based BCIs, the threshold criterion was a traditional way to detect the IC and NC states [9]–[11]. Because of the variability of EEG signal, it was difficult to achieve a high TPR in IC state and a low FPR

in NC state using a fixed threshold in conventional asynchronous SSVEP-based BCIs. So a novel pseudo-key-based approach was proposed [12]. Other researchers tried to use different algorithms such as k-nearest neighbors (KNN) [13], cluster analysis of canonical correlation analysis (CCA) coefficients [14] and so on to improve the performance of the asynchronous SSVEP-based BCI. More recently, support vector machine was employed to detect asynchronous SSVEP BCI with mixed-coded visual stimuli [15]. All of these researches were utilized traditional machine learning techniques and chose flicker as the stimulus.

Recently, deep learning, as a sub-field of machine learning, has made impressive advances in solving real-world problems such as computer vision and natural language processing [16]. Convolutional neural networks (CNN) is one supervised learning approach for deep leaning. And it was first proposed by Fukushima in 1988 [17]. Then Lecun *et al.* applied a gradient-based learning algorithm to CNNs and obtained successful results for the handwritten digit classification problem [18]. After that, researchers further improved CNNs and reported state-of-the-art results in many recognition tasks. And AlexNet [19], GoogLeNet [20], ResNet [21] were proposed to do deal with the ImageNet task. And ResNet-152 shows only 3.57% error, which was better than human errors for this task at 5% [22]. However, applications of deep neural nets in EEG signal detection are still at the beginning stage. Stober *et al.* combined CNN with DLSVM output layers to achieve rhythm classification among individuals [23]. Cecotti developed a 4-layer CNN to achieve the detection of P300 [24]. And CNN was also used to classify synchronous SSVEP [25], [26]. However, no research has tried to use CNN to realize asynchronous SSVEP-BCI or SSMVEP-BCI.

Besides, BCI is highly appealing in rehabilitation as it is possible to obtain patients' volitions, such that their volitions can be explicitly utilized to provide proper feedback. And the interaction between the rehabilitation process and the patient can induce proper plasticity. Actually, there were several studies applying SSVEP/SSMVEP BCI into rehabilitation. An EEG-driven lower limb rehabilitation training system was proposed in our precious study [27]. Virtual reality, BCI and robot are introduced into the system. A motion stimulus, which was different from the one proposed in the current study but induced similar SSMVEP response, were reported. When users were visually engaged at the motion stimulus, this system could automatically support corresponding visual and somatosensory feedback. However, the average accuracy using CCA to detect NC and IC was only 81.4%. For another more recent study, a BCI-based action observation rehabilitation game, a flickering action video, was used to supposedly activate the mirror-neuron system [28]. SSVEP was used to classify whether or not the participant was watching the stimulus video, i.e. a two-class scenario: NC and IC. And the results showed that the BCI interaction condition (providing rewards based on the classification) could active significantly stronger mirror-neuron system than the condition without BCI feedback. While the average classification accuracy was only 82.02 $\pm$ 8.72% for healthy participants. The higher accuracy, the more precise feedback could be provided. Thus a CNN-based method was proposed to improve the detection accuracy of asynchronous SSMVEP, which could move the BCI based rehabilitation forward.

In this study, we proposed a novel deep neural network to achieve asynchronous SSMVEP-BCI for the first time. The features of SSMVEP were mainly reflected in frequency domain. So the frequency features of EEG data were chosen as the proposed CNN input. Further, two processing pipelines were designed. The first one was using CNN to discriminate NC and all the states in IC at the same time (FFT-CNN). The second one was using CNN to discriminate IC and NC state and CCA to perform IC states classification when IC was detected (FFT-CNN-CCA). In addition, batch normalization (BN) was induced to improve the generalization ability of the network. Finally, the results were compared with two traditional algorithms for asynchronous SSVEP-BCI: threshold based on CCA (CCA-THD) and CCA followed by KNN (CCA-KNN).

## II. Methods

### A. Eeg Signal Measurement

EEG signals were recorded with a commercial research-grade EEG system (gUSBamp and Ladybird electrodes, g.tec Guger Technologies, Austria). Six electrodes were placed at PO3, PO4, POz, O1, O2, and Oz of the international 10 – 20 system. Left or right earlobe was used as the reference and Fpz was used as ground. All electrodes' impedances were kept below 5 kOhm following the guideline of the manufacturer. The sampling frequency was 1200 Hz. The signals were band-pass filter between 1 and 100 Hz and a notch filter from 48 Hz to 52 Hz was used to eliminate the power line interface.

### B. Experiment Protocol

Ten healthy male participants (aged from 23 to 27) and three stroke patients (aged from 30 to 70;diagnosed by computed tomography or MRI between 2 weeks and 6 months following stroke onset; no psychiatric disorder or expressive apraxia) participated in the experiments. The experimental protocol was approved by the Institutional Review Board of Xi'an Jiaotong University, China and the Research Ethics Committee of Xijing Hospital, China. Written Informed Consent forms were obtained from healthy participants and from the legal guardians of stroke patients before their participation in the experiments.

During the experiment, the participants were seated in a comfortable chair and were briefed on the tasks to be performed. The participants were asked to watch the computer screen on which the visual dues and feedback information were displayed. And they were asked to avoid performing any sudden jerking movements during the experiment.

Figure 1 illustrated the chessboard motion paradigm for inducing SSMVEP, which was proposed in our previous study [29]. In the chessboard paradigm, each ring was divided into black and white lattices of equal numbers and sizes. Thus, the total areas of the bright and dark regions in each

Fig. 1. The chessboard motion paradigm for inducing SSMVEP.

ring were always identical. The multiple rings in the chessboard contracted as the phase of the sinusoid signal of the targeted frequency changed from 0 degrees to 180 degrees and expanded as the phase changes from 180 degrees to 0 degrees. As shown in our previous study [4], [29], the periodic motions of the chessboard could elicit SSMVEP in EEG from the occipital area when the participants gazed at the chessboard. In the current study, four targets were displayed on the screen with motion frequencies of 7.5 Hz, 8.57 Hz, 9.23 Hz, and 8 Hz in the left, right, up, and down position of the screen, respectively.

For healthy participants, at the beginning of each trial, four letters ('L', 'R', 'U', 'D') would appear at the screen for 2 seconds, at the left, right, up, and down edges of the monitor, respectively. And one of the four letters was green while the other three were yellow. The green letter indicated the position of the target for the trial, at which participant would then engage his or her gaze for the remainder of the trail. At the beginning of the 3rd second, the four chessboards would replace the four letters, appearing on the screen for a duration of 6 seconds, during which the motions of the chessboards were modulated at the four frequencies stated above. The participants were asked to gaze at the target identified by the green letter (shown during 0 - 2 seconds) for the entire 6 seconds duration of the trial. This was followed by a relaxation period of 4 seconds, during which the participant could relax the gaze. Then the next trial would begin, in which the participants would be instructed to gaze at the next target. A total of 20 trials were performed in one run, in which each target was repeated for five times. A longer break was given between two runs, and a total of two runs were collected from each healthy participant.

For stroke patients, the protocol was similar to that of the healthy participants. The differences were 1) the chessboard appeared for a duration of 5 seconds; 2) the break periods between trials were 3 seconds long; 3) each target was repeated for four times in one run; 4) a total of four runs of data were collected for each stroke patient. These changes were to account for the potentially shorter attention span of stroke patients.

All EEG data and event time stamps (the beginning and end of each trial) were recorded by PC and processed offline.

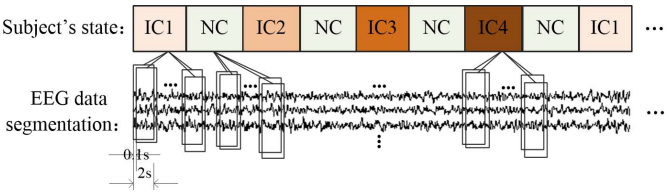


Fig. 2. The segmentation of EEG data.

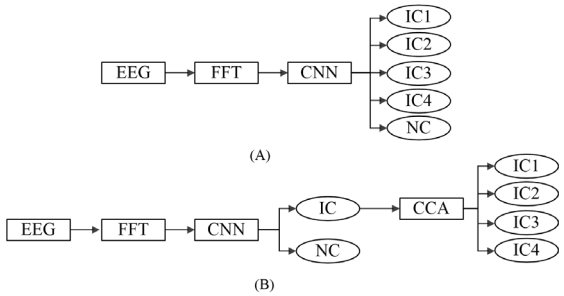


Fig. 3. The schematic diagram of the two CNN pipelines. (A) The FFT-CNN pipeline. (B) The FFT-CNN-CCA pipeline.

### C. Input Data Conditioning

The acquired EEG data were preprocessed by a band-pass filter from 3 Hz to 40 Hz. According to the time stamps, the EEG data were segmented into four IC groups and NC groups, as shown in Figure 2. IC groups included four states: IC1, IC2, IC3, and IC4 which responded to the Left, Right, Top, and Bottom chessboard, respectively. Each group comprised of multiple 6-second epochs of EEG signals. Each epoch was further segmented using a 2-second sliding window with an overlap of 1.9 seconds.

As for the CNN method, each 2-second window was transformed into its frequency domain representation by Fast Fourier transform. The 60 frequency points between 3 Hz and 33 Hz were chosen as the input data of the network. This was performed for each of the six channels of the EEG data. Hence the size of the unit in input data to the CNN was $60 \times 6$ (frequency points $\times$ channels). Because of the segmentation described above, the total number of samples for each participant was $R \times (2 \times N) \times (\frac{T-2}{0.1}+1)$ (Healthy participants: R= 2, N = 20, T = 6; stroke patients: R= 4, N = 16, T = 5).

### D. Two Cnn Pipelines

As described briefly in the Introduction, two CNN pipelines were proposed in the study, the schematic diagram of which was illustrated in Figure 3. The first pipeline used a CNN as a multi-class classifier to distinguish between all states of the IC and NC state. The second strategy used a CNN to discriminate between IC and NC states, followed by a CCA operator to classify of the specific class within the IC states.

### E. Architecture of the Convolutional Neural Network

For both pipelines, the CNN consisted of six sequential layers as shown in Figure 4. The input data were preprocessed

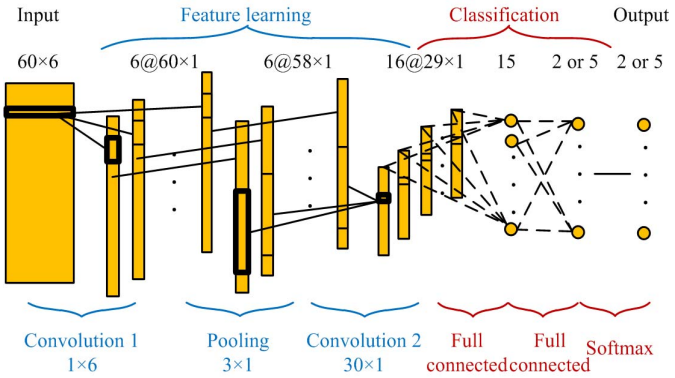


Fig. 4. Illustration of proposed CNN architecture.

as described in the previous section. $L_1$ and $L_3$ were both 2-D convolutional layer with Batch Normalization. The size and the number of the convolutional kernels were different in these two layers. The Rectified Linear Unit (ReLU) was used as the activation function in these two layers. Layer $L_2$ was a pooling layer that performed max pooling and down-sampling. And these layers effectively performed data-driven feature extraction from the data. The subsequent layers of the CNN shifted to performance classification. Layers $L_4$ and $L_5$ were fully-connected layers with dropout. The last layer, $L_6$, used the softmax function. And the loss function for classification was cross entropy for $k_c$ mutually exclusive classes.

Let $I$ denote the input tensor, the computations performed by each layer were described below:

— For $L_1$ (Convolution 1):

$$\sigma_k^1 = \sum_{i=1}^{i \le N_c} I_k(i) \times w_k^1(i) + b_k^1 \tag{1}$$

where $w_k^1(i)$ is a set of weights with $1 \le i \le N_c = 6$, $b_k^1$ is a bias, $N_c$ is the total number of EEG channels (6 in the current study), and $k$ indexes the feature map with $1 \le k \le N_k$. In this layer, there are a total of $N_k$ feature maps and the size of the convolutional kernel is $1 \times N_c$. In the current design, $N_k$ was equal to $N_c$. The output of each $\sigma_k^1$ was a $SF \times 1$ vector, where $SF$ denoted the samples in frequency domain and batch normalization was performed on this output. The output of batch normalization was passed through the ReLU activation function which was defined as:

— For $L_2$ (Pooling):

$$a_k^1 = g(BN(\sigma_k^1)) \tag{2}$$

where $g(x) = max(x, 0)$. This enables a faster convergence during the training stage [30]. The role of BN layers would be detailed in the subsequent sections. The output of each layer is denoted as $a_k^l$, where $l$ denotes the layer and $k$ denotes the feature map. Therefore, the output $a_k^1$ is fed into the $L_2$ max-pooling layer and is defined as:

— For $L_2$ (Pooling):

$$a_k^2 = mp(a_k^1) \tag{3}$$

where $mp(\cdot)$ is the max-pooling function. The size of the max-pooling kernel was $3 \times 1$ and the stride was 1. In this approach, the highest value was selected from the $3 \times 1$ patches of the feature maps. Therefore, this layer performs the down-sample operation on the input maps. This had been shown to alleviate the problem of overfitting caused by high dimensionality.

— For $L_3$ (Convolution 2):

$$\sigma_p^3(j) = \sum_{k=1}^{k \le N_k} \sum_{i=1}^{i \le N_t} (a_k^2(i + j - 1) \times w_p^3(j) + b_p^3(j)) \tag{4}$$

$$a_p^3 = g(BN(\sigma_p^3)) \tag{5}$$

where $p$ is the index of the convolutional feature map. This layer is similar to $L_1$. Both layers are followed by batch normalization. The output of batch normalization is passed through the ReLU activation function. The convolutional kernels had a size of $30 \times 1$ with a stride of 1. A total of 16 feature maps were in this layer.

— For $L_4$ (Full connected back-prop):

$$r^4 \sim \text{Bernoulli}(\lambda^4) \tag{6}$$

$$a^4 = r^4 \times a^3 \times \mathbf{W}^4 + \mathbf{b}^4 \tag{7}$$

where $\mathbf{W}^4, \mathbf{b}^4$ are the weights and biases, $\lambda^4$ is the scalar dropout rate. Dropout was implemented in this layer to help preventing overfitting [31]. The dropout randomly sets input elements to zero with a given probability. And the probability was 50% in this study. Moreover, each unit of $L_4$ was connected to each unit of $L_3$. The number of units in the output was 15.

— For $L_5$ (Full connected back-prop):

$$r^5 \sim \text{Bernoulli}(\lambda^5) \tag{8}$$

$$a^5 = r^5 \times a^4 \times \mathbf{W}^5 + \mathbf{b}^5 \tag{9}$$

where $\mathbf{W}^5, \mathbf{b}^5$ are the weights and biases, $\lambda^5$ is the scalar dropout rate. This layer is similar with $L_4$. And each unit of $L_5$ is connected to each unit of $L_4$. For generalization, the network uses two densely connected layers instead of one. The number of units in the output of this layer is equal to the number of classes in the input data.

— For $L_6$ (Output):

$$output = \frac{exp(a_r^5)}{\sum_{j=1}^{N_o} exp(a_j^5)} \tag{10}$$

where the softmax function is used to provide the classification output. The number of units in the output layer is equal to the number of classes. As such, for the FFT-CNN-CCA pipeline, this layer contains 2 units, representing IC and NC states. For the FFT-CNN pipeline, this layer contains 5 units, representing the IC1, IC2, IC3, IC4, and NC states. The details of these two strategies were provided in Section II.D.

### F. Training Parameters

The network weights were learned based on the stochastic gradient descent learning algorithm which used the standard error back propagation to optimize network weights. The cross-entropy function was used as the loss function. The learning rate was set at 0.01. The number of training epoch was set as 30, and the size of mini-batch for stochastic gradient descent was set to 8.

### G. Batch Normalization

Batch normalization reduces the internal covariance within the input samples, which means the inputs are linearly transformed to have zero mean and unit variance [22]. In section III.C, we would discuss the importance of batch normalization. The algorithm of BN was described below.

Outputs:

$$y_i = BN_{\gamma,\beta}(x_i) \tag{11}$$

Mini-batch mean:

$$\mu_\beta \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \tag{12}$$

Mini-batch variance:

$$\sigma_\beta^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_\beta)^2 \tag{13}$$

Normalize:

$$\hat{x_i} \leftarrow \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + \epsilon}} \tag{14}$$

Scaling and shifting:

$$y_i = \gamma\,\hat{x_i} + \beta \equiv BN_{\gamma,\beta}(x_i) \tag{15}$$

### H. Traditional Algorithms to Realize Asynchronous Ssvep-Bci

The illustrate the advantage of the two proposed CNN-based algorithms, we compared their performance with two traditional SSVEP-BCI algorithms that have been widely reported in the literature.

*1) Threshold-Based on Cca (Cca-Thd):* CCA is widely used in SSVEP processing. It calculates the correlations between reference signals and multi-channel EEG signal [32]. In these previous studies, CCA was successfully applied in targets recognition among different IC states. However, CCA has not been shown to be able to detect IC states directly, which is required to operate an asynchronous SSVEP-BCI. To the end, the threshold method based on CCA coefficients was proposed to realize asynchronous SSVEP-BCI [13]. In this study, the entropy features based on CCA coefficients were used for threshold calculation. The entropy H was defined as:

$$H = -\sum_{i=1}^{4} \rho_i \times log(\rho_i) \tag{16}$$

where the correlation coefficient using CCA was defined as:

$$\rho_i = max_{w_x,w_y} \frac{\mathrm{E}[w_x^T X Y^T w_y]}{\sqrt{\mathrm{E}[w_x^T X X^T w_x]\mathrm{E}[w_y^T Y Y^T w_y]}} \tag{17}$$

The thresholds were choosing among 1.82, 1.88, 1.94, 2.00, and 2.06.The best threshold for each participant was selected based on the highest accuracy.

*2) K-Nearest Neighbors Based on Cca (Cca-Knn):* K-nearest neighbors algorithm was proposed as an alternate method for operating an asynchronous SSVEP-BCI using CCA. The set of CCA coefficients were used as the features, from which KNN is used to classify different states. A given test sample was assigned to the class based on the Euclidean distance to the Kth nearest neighbor. CCA-KNN showed a slightly higher accuracy than the threshold method based on CCA coefficients [13]. In this study, CCA-KNN classifier was trained for $K = 5$, 55, 105, 155, and 205, respectively, and $K$ was selected on the test set to achieve the best accuracy.

### I. Quantification of Classifiers' Performance

FPR, TPR, and Accuracy (ACC) were used as the metrics to qualify the performance of the asynchronous SSVEP-BCI. They were computed for the two proposed methods and for the two traditional algorithms. FPR, TPR, and ACC were defined as follows [33]:

$$FPR = \frac{FP}{FP + TN} \tag{18}$$

$$TPR = \frac{TP}{TP + FN} \tag{19}$$

$$ACC = \frac{TP + TN}{TP + FN + FP + TN} \tag{20}$$

where $FP$, $TN$, $TP$, and $FN$ are the number of false positive, true negative, true positive, and false negative, respectively. Positive refers to the IC states and negative refers to the NC state in our study.

A 5-fold cross-validation scheme was performed for healthy participants' data. All the two runs' data were partitioned into five equal-sized subsamples sequentially in time. Of the five subsamples, a single subsample was retained as the validation data for testing the model, and the remaining four subsamples were used as training data. There was no overlapping part in both training and test subsets. The cross-validation process was then repeated five times, with each of the five subsamples used exactly once as the validation data. For stroke patients' data, 4-fold cross-validation was performed with a similar approach to that of the healthy subjects' analysis.

### J. Statistical Analysis

The mixed effect model of Analysis of variance (ANOVA) was used for statistical analysis. Methods ("1": CCA-THD, "2": CCA-KNN, "3": FFT-CNN, "4": FFT-CNN-CCA) and groups (mentioned in III.A) were used as fixed factors and participants were used as random factors. FPR, TPR, and ACC values were the response variables. The Bonferroni post hoc analysis was used to assess significance. The statistical significance level was 0.05.

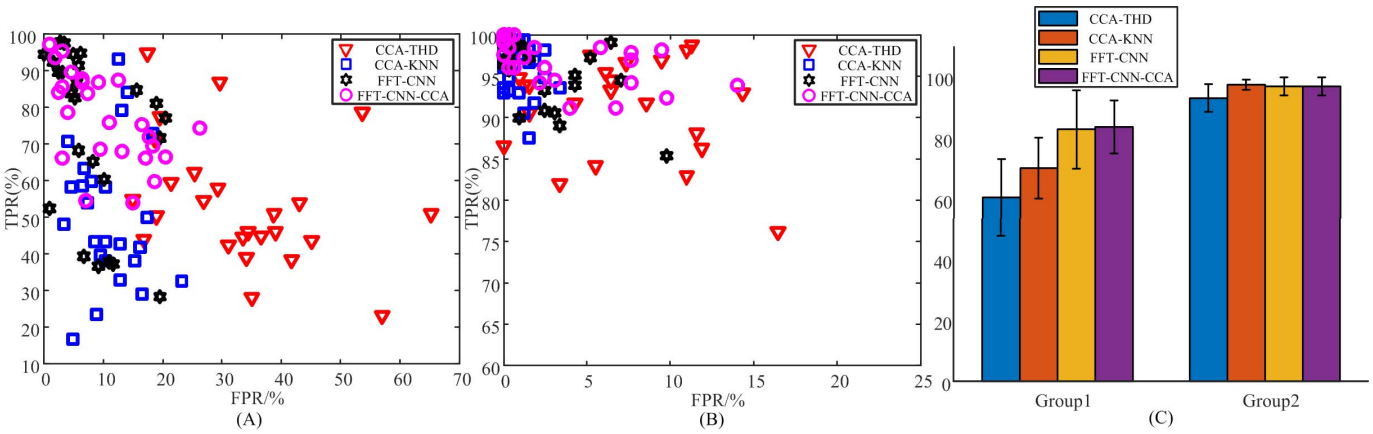


Fig. 5. The comparisons of the FPR, TPR, and ACC among the four different algorithms ON healthy participants' data. (A) The comparison of FPR and TPR in group 1. (B) The comparison of FPR and TPR in group 2. (C) The comparison of ACC in group 1 and group 2. (Please note that the domains of coordinates axis in (A) and (B) are not the same).

TABLE I
THE ACCURACIES USING DIFFERENT THRESHOLDS IN CCA-THD METHOD AND USING DIFFERENT *K* IN CCA-KNN METHOD

| | CCA-THD (%) | | | | | CCA-KNN (%) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 1.82 | 1.88 | 1.94 | 2.00 | 2.06 | 5 | 55 | 105 | 155 | 205 |
| S1 | 45.43 | 43.17 | 43.08 | **47.29** | 47.23 | 54.82 | 60.15 | **60.27** | 60.24 | 60.12 |
| S2 | 78.84 | **87.68** | 86.25 | 77.44 | 60.40 | 96.28 | **96.59** | 96.43 | 94.22 | 96.16 |
| S3 | 80.52 | **90.43** | 89.63 | 81.46 | 63.48 | 95.82 | **96.49** | 96.19 | 96.01 | 95.82 |
| S4 | 81.52 | 91.46 | **95.37** | 92.13 | 76.25 | 95.58 | 96.19 | **96.43** | 96.31 | 96.25 |
| S5 | 50.73 | 54.76 | 62.5 | **70.61** | 66.25 | 76.19 | **78.63** | 77.93 | 77.87 | 77.32 |
| S6 | 52.23 | 56.13 | 61.77 | **66.77** | 60.30 | 70.79 | **73.81** | 72.74 | 72.5 | 71.83 |
| S7 | 48.66 | 47.71 | 48.99 | **51.25** | 47.80 | 56.25 | **60.49** | 59.73 | 59.21 | 59.15 |
| S8 | 78.35 | 91.40 | **95.09** | 91.13 | 73.78 | 98.32 | 98.35 | 98.38 | **98.45** | 98.32 |
| S9 | 47.99 | 49.91 | 57.23 | **65.82** | 64.60 | 74.42 | 76.46 | **76.52** | 76.34 | 75.67 |
| S10 | 81.49 | 92.62 | **96.00** | 92.35 | 74.79 | 97.59 | 98.02 | 98.38 | 98.45 | **98.48** |

## III. RESULTS

### A. Healthy Participants' Data

For an asynchronous SSMVEP-BCI system, one needs to classify IC and NC states, as well as classify all the sub-states within the IC states. For the CCA-THD method, the FPR, TPR, and ACC were calculated based on all the thresholds mentioned in Methods. Then we selected the best threshold for each participant based on the highest ACC as shown in Table 1. Finally, 2.00, 1.88, 1.88, 1.94, 2.00, 2.00, 2.00, 1.94, 2.00, and 1.94 were selected as the thresholds for the ten participants respectively. Furthermore, the accuracies of CCA-KNN classifier with different $K$ were also shown in Table 1. And 105, 55, 55, 105, 55, 55, 55, 155, 105, and 205 were selected as the best $K$ for each participant.

Based on the CCA-THD results, all the healthy participants were separated into two groups (Group 1 and Group 2). Group 1 consists of subject 1, 5, 6, 7, and 9, whose accuracies of CCA-THD were lower than 80%. The other participants were in group 2. Figure 5 showed the comparisons of FPR, TPR, and ACC among the four different methods based on 5-fold cross-validation results. As shown in Figure 5(A), even though we chose the best threshold for the CCA-THD method and best $K$ for the CCA-KNN method for each participant in Group1, it was still difficult to achieve high TPR and low FPR at the same time. The TPR, FPR, and ACC of the CCA-THD method were 53.77% ± 17.19%, 33.07% ± 13.13%, and 60.35% ± 12.61%, respectively. The TPR, FPR, and ACC of the CCA-KNN method were 50.89% ± 19.12%, 11.00% ± 5.01%, and 69.95% ± 9.98%, respectively. Similar to the report in the literature [13], CCA-KNN consistently outperformed CCA-THD in ACC. On the other hand, the proposed CNN-based methods showed a significant performance improvement. The TPR, FPR, and ACC of the FFT-CNN pipeline were 73.44% ± 22.50%, 8.10% ± 6.24%, and 82.67% ± 12.84% respectively. The TPR, FPR, and ACC of the FFT-CNN-CCA pipeline were 77.12% ± 12.42%, 10.20% ± 6.97%, and 83.46% ± 8.70% respectively. For both TPR and FPR, the two CNN pipelines provided marked improvements over CCA-THD. While CCA-KNN's FPRs were similar to those of the two CNN-based methods, its TPR was in general lower than the two CNN-based methods. For Group2, shown in Figure 5(B), with high baseline performance (ACC>80% using CCA-THD), the performance gain by the CNN-based algorithms were not as clear as in Group1. The TPR of CCA-THD, CCA-KNN, FFT-CNN, and FFT-CNN-CCA were 92.10% ± 6.10%, 95.52% ± 2.92%, 95.67% ± 3.89%, and 96.83% ± 2.77%, respectively. The FPR of these four methods were 6.27% ± 4.86%, 0.95% ± 0.93%, 2.16% ± 2.67%, and 3.41% ± 4.00%, respectively. The above observation

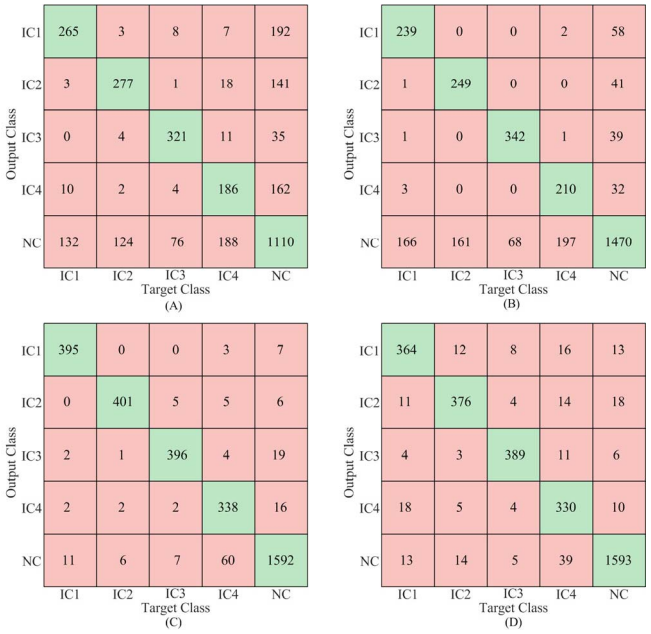


Fig. 6. The classification confusion matrices for subject 9. (A) CCA-THD method. (B) CCA-KNN method. (C) FFT-CNN pipeline. (D) FFT-CNN-CCA pipeline.

is confirmed in Figure 5(C), where the ACC value by the two proposed CNN pipelines showed a marked increase, with $>$ 10% for the participants in Group 1, while the changes in ACC values were less for Group 2.

To further quantify the differences between the proposed methods(Method 3: FFT-CNN, 4: FFT-CNN-CCA) and traditional methods (Method1: CCA-THD, 2: CCA-KNN), we used mixed effect model of ANOVA on healthy participants' data. Based on the conditional residual for accuracy, two outliers were removed and the remaining accuracies fit a normal distribution. The interaction of Methods and Groups was statistically significant on ACC values. According to the Bonferroni Pairwise comparison tests, the ACC values of Method 2 were significantly higher than Method 1 ($p < 0.001$), which was in line with a previous study in [13]. The ACC values of the both Methods 3 and 4 were significant higher than the ACC values using Methods 1 and 2 (all $p < 0.001$), and the ACC values of Method 4 and Method 3 had no significant difference ($p = 1$). Further, Bonferroni Pairwise comparisons revealed the interactions between Methods and Groups. In Group 1, the ACC values of Methods 3 and 4 were both significant higher than the ACC values of Methods 1 and 2 (all $p < 0.001$). However, in group 2, there was no significant difference in the ACC values of Method 2 and Method 1 ($p = 0.041$), Method 3 and Method 1 ($p = 0.140$), Method 4 and Method 1 ($p = 0.156$), Method 3 and Method 2 ($p = 1$), Method 4 and Method 2 ($p = 1$). Thus, the proposed neural network architecture provides a significant performance improved for participants with low performance (Group 1), while no such effect was found for participants with higher performance (Group 2).

The performance advantages of the algorithms could be seen in better detail in Figure 6, where the classification confusion matrices of the sum of the 5-fold cross-validation results from Participant 9 (in Group 1) was shown. The values in the green background (diagonal entries) were the number of correct classifications, and the values in the red background (off-diagonal entries) were the number of misclassifications. The total number of ICs were the same as the total number of NC. Figure 6(A) and 6(B) was the results from CCA-THD and CCA-KNN, respectively. There were more misclassifications between NC and IC when compared to the misclassifications between all the IC states: the values in the last row and the last column in the red background were significantly larger than other values in red background. Thus, the low performance of the traditional algorithms could be attributed to the large number of misclassifications between the IC and NC states. In contrast, Figure 6(C) and Figure 6(D) showed the results of the two proposed CNN-based methods. Evidently, the proposed method significantly reduced the misclassifications between the NC and IC states.

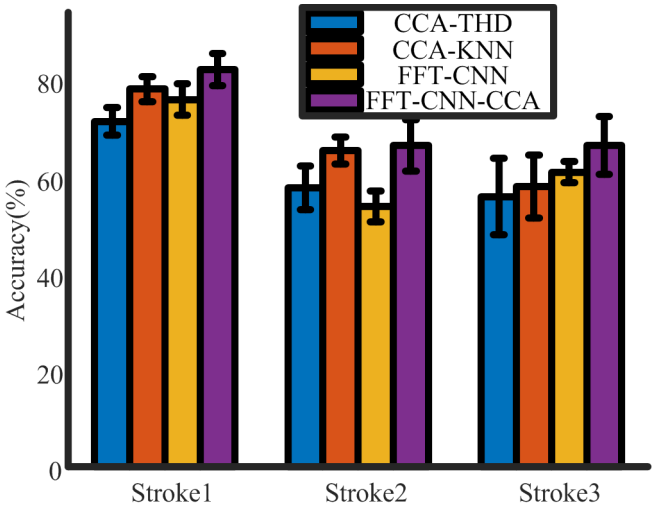


Fig. 7. The comparisons of ACC values between the four different algorithms ON stroke patients' data.

## B. Stroke Patients' Data

The performance of the proposed methods was further evaluated on the EEG data collected from three stroke patients. The best threshold for CCA-THD method and $K$ for CCA-KNN method were selected in the same method as the healthy participants. The selected CCA-THD thresholds for the stroke patients were 1.88, 1.88, and 2.06, respectively. And the best $K$ values for the CCA-KNN method were 55, 55, and 55 for each stroke participant, respectively. Figure 7 presented the ACC comparisons of the four methods. The ACC values of the three patients using CCA-THD method were 71.45% $\pm$ 2.84%, 57.71% $\pm$ 4.55%, and 55.90% $\pm$ 7.85%, respectively. The baseline performance (CCA-THD) of the three patients were all lower than 80% in ACC. This is similar to the baseline performance level of Group 1 of the healthy participants. CCA-KNN also outperformed CCA-THD for all stroke patients. The ACC of the three patients using FFT-CNN-CCA method were 82.16% $\pm$ 3.29%, 66.43% $\pm$ 5.35%, and 66.48% $\pm$ 5.98%, respectively. Evidently, the FFT-CNN-CCA pipeline offered a significant performance improvement of CCA-THD method, with $>$ 10% in ACC for all three patients. On the other hand, the FFT-CNN-CCA offered approximately 5%

TABLE II
THE ACC VALUES USING FFT-CNN AND FFT-CNN-CCA WITH AND WITHOUT BN LAYERS

| | FFT-CNN (without BN) | FFT-CNN (with BN) | FFT-CNN-CCA (without BN) | FFT-CNN-CCA (with BN) |
|---|---|---|---|---|
| | ACC(%) | ACC(%) | ACC(%) | ACC(%) |
| S1 | 65.00 | 77.29 | 73.72 | 75.09 |
| S2 | 68.08 | 97.04 | 93.41 | 95.95 |
| S3 | 78.11 | 95.67 | 72.56 | 94.89 |
| S4 | 86.83 | 92.80 | 91.19 | 93.84 |
| S5 | 66.68 | 94.21 | 80.67 | 90.88 |
| S6 | 77.20 | 84.54 | 60.91 | 82.13 |
| S7 | 55.46 | 62.13 | 67.23 | 76.16 |
| S8 | 99.27 | 99.43 | 99.36 | 99.54 |
| S9 | 70.37 | 95.18 | 69.39 | 93.05 |
| S10 | 89.21 | 99.30 | 79.54 | 99.33 |
| Average | 75.62 ± 13.17 | 89.76 ± 11.91 | 78.80 ± 12.48 | 90.09 ± 9.05 |

improvement over the ACC values of CCA-KNN method for all three patients. However, the FFT-CNN pipeline only showed performance improvement over the two traditional methods for Stroke 3, where the ACC of method 1, 2, 3, and 4 were 55.90% ± 7.85%, 57.99% ± 6.51%, 60.94% ± 2.16%, and 66.48% ± 5.98%, respectively. For Stroke 1, FFT-CNN's performance was higher than CCA-THD, but lower than CCA-KNN. For Stroke 2, its performance was the lowest among the four methods. The FPR in this case was 11.85% ± 3.32%, while the TPR was only 19.56% ± 3.40%. This could be attributed to the fact that the model overfitting the training data for Stroke 2.

### C. Role of Batch Normalization Layers

The previous studies in the literature using CNN to classify synchronous SSVEP [25], [26], [34] did not discuss the role of the BN layer. To test the effect of the BN layers in the network for asynchronous SSMVEP, the network was trained without the BN layers under the same conditions in this study. The accuracies were presented in Table 2. The accuracy using FFT-CNN method without BN was only 75.62%. And the accuracy using FFT-CNN-CCA method without BN was only 78.8%. The accuracies using the model without BN were lower than the accuracies using the model with BN. Thus, we concluded that BN is an essential component of the proposed pipelines.

## IV. DISCUSSION AND CONCLUSION

These results indicate that 1) the proposed CNN-based methods are capable of detecting the NC and all the IC states, and 2) provide a significant advantage for participants with low baseline performance when using the traditional methods. Furthermore, CCA-KNN showed a slightly better performance than CCA-THD in the healthy subject's data, which was in line with the study in [13].

Since CNN has the property of automatic feature extraction. Thus we can also use the raw EEG data without FFT as the input data of the neural network. However, the input data of the neural network in this study were the frequency features of EEG data. Because the characterizes of SSVEP mainly response in the frequency domain. And the size of the unit of input data was small (60 × 6). If the time domain was choosing as the input unit, the size would be 2400×6. So the training time would be longer. The runtime of training for the FFT-CNN-CCA method was only about 17 s. So the frequency features of EEG data were chosen as the input of CNN.

Furthermore, the time length is one of the most important parameters for SSVEP-BCI, as it influences the speed of the system. 2s windows has been chosen in many studies, for example [9], [13], and longer window lengths were also considered in other asynchronous SSVEP-BCI studies [12], [15]. Thus, we chose 2s time windows to ensure a sufficiently fast system response time.

It was evident that stroke patients' data showed poor classification accuracy compared with healthy participants' data. This poor performance could due to poorer concentration of the stroke patients and excessive noise in their data. Actually, during the experiments, the stroke patients could not focus the stimuli for a long time. And it was difficult for them seating there quietly without any sudden jerking movements. Thus the classification accuracy in stroke patients was in general lower than healthy participants. However, we demonstrated that the FFT-CNN-CCA pipeline offered significantly better performance than traditional methods in stroke patients.

Currently, our model focused on offline classification for asynchronous SSMVEP-BCI. However, for practical applications, the online classification performance needed to be assessed. One of the major factors to be considered while deploying CNN in real-time was the time taken for decision making. To evaluate this, the structure and weights of the trained neural network were saved, and tested with the unseen test data. All experiments were carried out on an Intel i5 4460 CPU with Windows 8 OS without GPU acceleration. The runtime between input of test data and the decision output was recorded. The average time elapsed for a single decision was 0.0041s which was well within the acceptable limit for the online deployment, hence making CNNs a suitable candidate for asynchronous SSMVEP-BCIs. So future work will be conducted to: 1) develop a real-time asynchronous SSMVEP-BCI system using CNN, 2) recruit more stroke patients and collect more data to increase the size of the input to check whether the performance of the proposed algorithms could be further improved, and 3) gain more insights into the internal workings of the network.

Finally, we concluded that the proposed CNN-based methods were suitable candidate for developing asynchronous SSMVEP-BCI systems for healthy participants and stroke patients. And the proposed methods can provide improved performance.

## ACKNOWLEDGMENT

The authors would like to thank the participants for participating in these experiments and anonymous reviewers for their helpful comments.

## REFERENCES

[1] J. Wolpaw *et al.*, "Brain-computer interface technology: A review of the first international meeting," *IEEE Trans. Rehabil. Eng.*, vol. 8, no. 2, pp. 164–173, Feb. 2000.

[2] F. B. Vialatte, M. Maurice, J. Dauwels, and A. Cichocki, "Steady-state visually evoked potentials: Focus on essential paradigms and future perspectives," *Prog. Neurobiol.*, vol. 90, pp. 418–438, Apr. 2010.

[3] X. Chen, Y. Wang, M. Nakanishi, X. Gao, T.-P. Jung, and S. Gao, "High-speed spelling with a noninvasive brain–computer interface," *Proc. Natl. Acad. Sci. USA*, vol. 112, no. 44, pp. E6058–E6067, Nov. 2015.

[4] W. Yan, G. Xu, J. Xie, M. Li, and Z. Dan, "Four novel motion paradigms based on steady-state motion visual evoked potential," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 8, pp. 1696–1704, Aug. 2018.

[5] G. Pfurtscheller and T. Solis-Escalante, "Could the beta rebound in the EEG be suitable to realize a "brain switch?"" *Clin. Neurophysiol.*, vol. 120, no. 1, pp. 24–29, Jan. 2009.

[6] S. He *et al.*, "A P300-based threshold-free brain switch and its application in wheelchair control," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 6, pp. 715–725, Jun. 2017.

[7] S. G. Mason and G. E. Birch, "A brain-controlled switch for asynchronous control applications," *IEEE Trans. Biomed. Eng.*, vol. 47, no. 10, pp. 1297–1307, Oct. 2000.

[8] Y. Li, J. Pan, F. Wang, and Z. Yu, "A hybrid BCI system combining P300 and SSVEP and its application to wheelchair control," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 11, pp. 3156–3166, Nov. 2013.

[9] H. Cecotti, "A self-paced and calibration-less SSVEP-based brain–computer interface speller," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 18, no. 2, pp. 127–133, Apr. 2010.

[10] G. R. Müller-Putz and G. Pfurtscheller, "Control of an electrical prosthesis with an SSVEP-based BCI," *IEEE Trans. Biomed. Eng.*, vol. 55, no. 1, pp. 361–364, Jan. 2008.

[11] R. C. Panicker, S. Puthusserypady, and Y. Sun, "An asynchronous P300 BCI with SSVEP-based control state detection," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 6, pp. 1781–1788, Jun. 2011.

[12] J. Pan, Y. Li, R. Zhang, Z. Gu, and F. Li, "Discrimination between control and idle states in asynchronous SSVEP-based brain switches: A pseudo-key-based approach," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 21, no. 3, pp. 435–443, May 2013.

[13] N.-S. Kwak, K.-R. Müller, and S.-W. Lee, "A lower limb exoskeleton control system based on steady state visual evoked potentials," *J. Neural Eng.*, vol. 12, 2015, Art. no. 056009.

[14] P. Poryzala and A. Materka, "Cluster analysis of CCA coefficients for robust detection of the asynchronous SSVEPs in brain–computer interfaces," *Biomed. Signal Process. Control*, vol. 10, pp. 201–208, Mar. 2014.

[15] K. Suefusa and T. Tanaka, "Asynchronous brain–computer interfacing based on mixed-coded visual stimuli," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 9, pp. 2119–2129, Sep. 2018.

[16] M. Liu, W. Wu, Z. Gu, Z. Yu, F. F. Qi, and Y. Li, "Deep learning based on batch normalization for P300 signal detection," *Neurocomputing*, vol. 275, pp. 288–297, Jan. 2018.

[17] K. Fukushima, "Neocognitron: A hierarchical neural network capable of visual pattern recognition," *Neural Netw.*, vol. 1, no. 2, pp. 119–130, 1988.

[18] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[20] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[22] M. Z. Alom *et al.* (Mar. 2018). "The history began from AlexNet: A comprehensive survey on deep learning approaches." [Online]. Available: https://arxiv.org/abs/1803.01164

[23] S. Stober, D. J. Cameron, and J. A. Grahn, "Using convolutional neural networks to recognize rhythm stimuli from electroencephalography recordings," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 1449–1457.

[24] H. Cecotti and A. Graser, "Convolutional neural networks for P300 detection with application to brain-computer interfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 3, pp. 433–445, Mar. 2011.

[25] H. Cecotti, "A time–frequency convolutional neural network for the offline classification of steady-state visual evoked potential responses," *Pattern Recognit. Lett.*, vol. 32, no. 8, pp. 1145–1153, Jun. 2011.

[26] N.-S. Kwak, K.-R. Müller, and S.-W. Lee, "A convolutional neural network for steady state visual evoked potential classification under ambulatory environment," *PLoS ONE*, vol. 12, no. 2, Feb. 2017, Art. no. e0172578.

[27] X. Zhang, G. Xu, J. Xie, M. Li, W. Pei, and J. Zhang, "An EEG-driven lower limb rehabilitation training system for active and passive co-stimulation," in *Proc. 37th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Aug. 2015, pp. 4582–4585.

[28] M. Tani *et al.*, "Action observation facilitates motor cortical activity in patients with stroke and hemiplegia," *Neurosci. Res.*, vol. 133, pp. 7–14, Aug. 2018.

[29] X. Zhang, G. Xu, J. Xie, and X. Zhang, "Brain response to luminance-based and motion-based stimulation using inter-modulation frequencies," *PLoS ONE*, vol. 12, no. 1, Nov. 2017, Art. no. e0188073.

[30] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 8609–8613.

[31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[32] Z. Lin, C. Zhang, W. Wu, and X. Gao, "Frequency recognition based on canonical correlation analysis for SSVEP-based BCIs," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 12, pp. 2610–2614, Dec. 2006.

[33] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006.

[34] V. Bevilacqua *et al.*, "A novel BCI-SSVEP based approach for control of walking in virtual environment using a convolutional neural network," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2014, pp. 4121–4128.